

图文 117 一个消费组中的多个Consumer是如何均匀分配消息队列的？

79 人次阅读 2020-04-08 07:00:00

[详情](#) [评论](#)

一个消费组中的多个Consumer是如何均匀分配消息队列的？

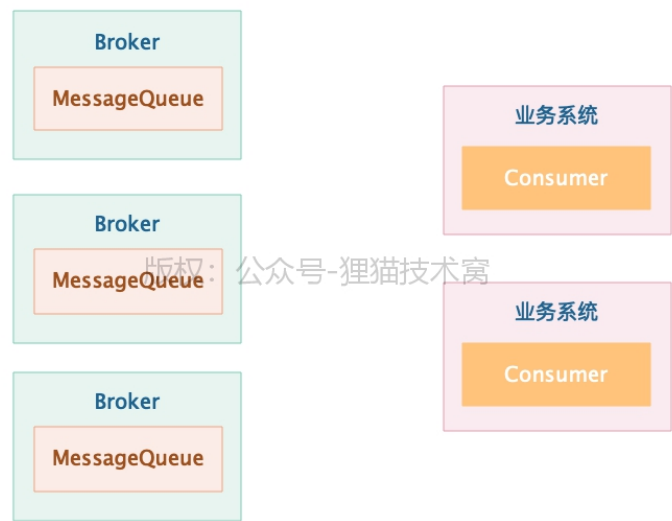


继《从零开始带你成为JVM实战高手》后，阿里资深技术专家携新作再度出山，重磅推荐：

(点击下方蓝字试听)

[《从零开始带你成为MySQL实战优化高手》](#)

今天来给大家讲解一下当你一个业务系统部署多台机器的时候，每个系统里都启动了一个Consumer，多个Consumer会组成一个ConsumerGroup，也就是消费组，此时就会有一个消费组内的多个Consumer同时消费一个Topic，而且这个Topic是有多个MessageQueue分布在多个Broker上的，如下图所示

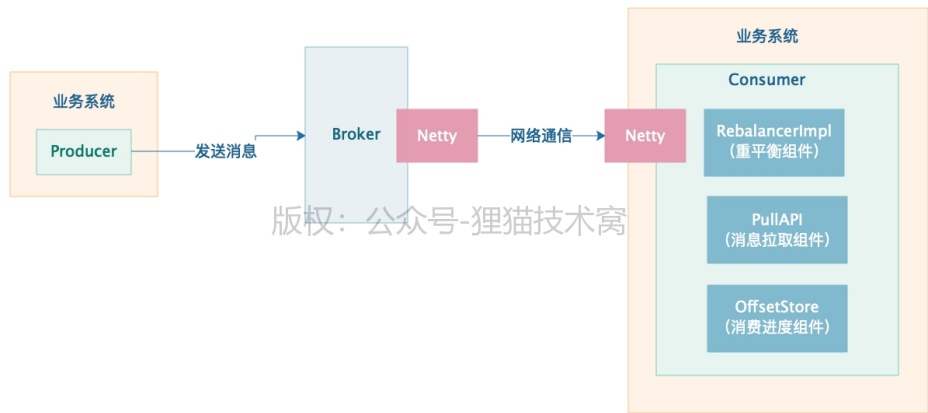


那么现在问题就来了，假设咱们一个业务系统部署在两台机器上，对应一个消费组里就有两个Consumer，那么现在一个Topic有三个MessageQueue，该怎么分配呢？

这就涉及到了**Consumer的负载均衡**的问题了。

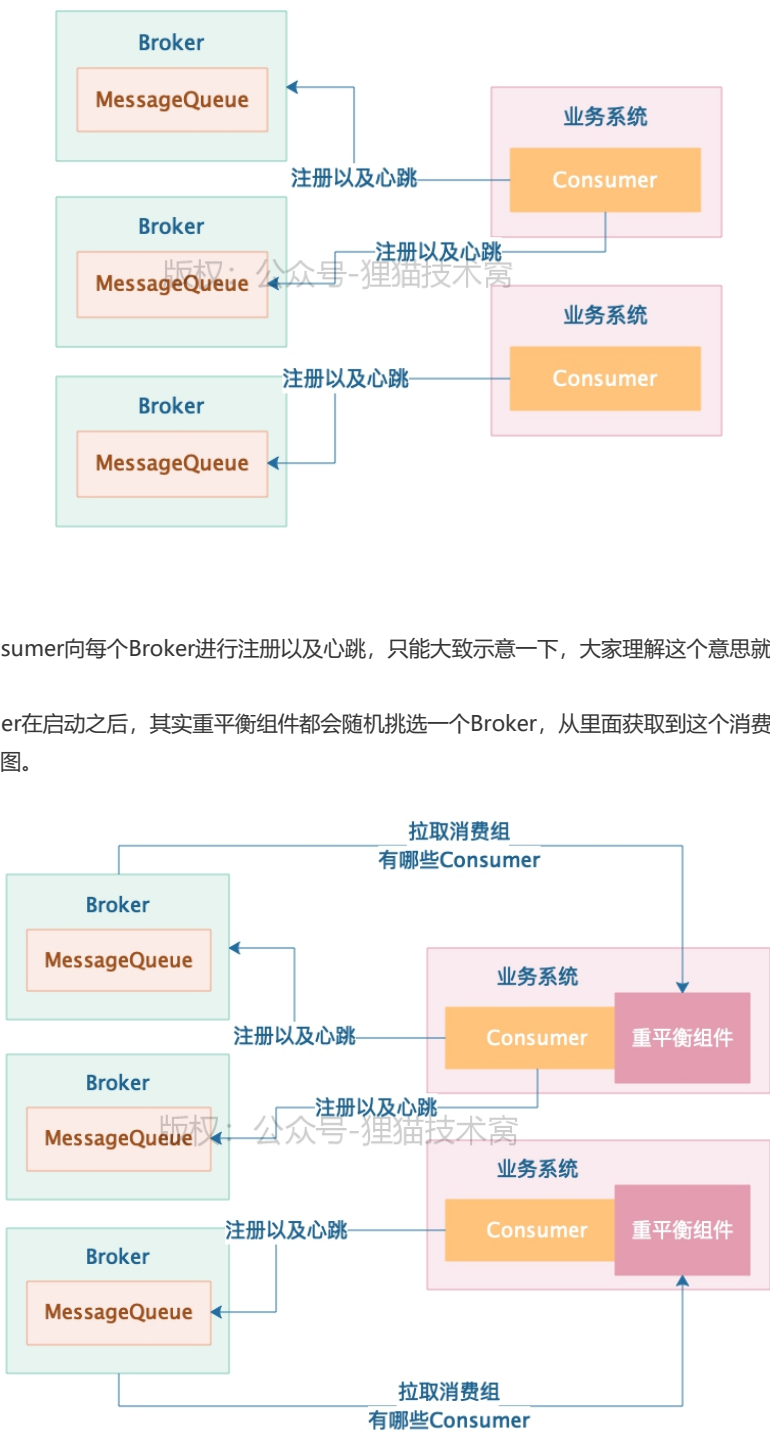
不知道大家是否还记得我们上一次讲Consumer启动的时候，讲到了几个关键的组件，分别是重平衡组件、消息拉取组件、消费进度组件

其实里面有一个Balancer重平衡组件，就是在这里专门负责多个Consumer的负载均衡的，如下图。



那么这个RebalancerImpl重平衡组件是如何将多个MessageQueue均匀的分配给一个消费组内的多个Consumer的呢？

实际上，每个Consumer在启动之后，都会干一件事情，就是向所有的Broker进行注册，并且持续保持自己的心跳，让每个Broker都能感知到一个消费组内有哪些Consumer，如下图。



上图里没法画出来Consumer向每个Broker进行注册以及心跳，只能大致示意一下，大家理解这个意思就好

然后呢，每个Consumer在启动之后，其实重平衡组件都会随机挑选一个Broker，从里面获取到这个消费组里有哪些Consumer存在，如下图。

此时重平衡组件一旦知道了消费组内有哪些Consumer之后，接着就好办了，无非就是把Topic下的MessageQueue均匀的分配给这些Consumer了，这个时候其实有几种算法可以进行分配，但是比较常用的一种算法就是简单的平均分配。

比如现在一共有3个MessageQueue，然后有2个Consumer，那么此时就会给1个Consumer分配2个MessageQueue，同时给另外1个Consumer分配剩余的1个MessageQueue。

假设有4个MessageQueue的话，那么就可以2个Consumer每个人分配2个MessageQueue了

总之，一切都是平均分配的，尽量保证每个Consumer的负载是差不多的。

这样的话，一旦MessageQueue负载确定了之后，下一步其实Consumer就知道自己要消费哪几个MessageQueue的消息了，就可以连接到那个Broker上去，从里面不停的拉取消息过来进行消费了，至于如何拉取消息消费，那就是下一次要讲的了。

讲完了拉取消息消费，我们再讲一下消费进度的管理，那么此时RocketMQ最最核心的底层原理就讲清楚了。

End

专栏版权归公众号**狸猫技术窝**所有

未经许可不得传播，如有侵权将追究法律责任

狸猫技术窝精品专栏及课程推荐：

- [《从零开始带你成为JVM实战高手》](#)
- [《21天互联网Java进阶面试训练营》（分布式篇）](#)
- [《互联网Java工程师面试突击》（第1季）](#)
- [《互联网Java工程师面试突击》（第3季）](#)

重要说明：

- 如何提问：每篇文章都有评论区，大家可以尽情留言提问，我会逐一答疑
- 如何加群：购买狸猫技术窝专栏的小伙伴都可以加入狸猫技术交流群，一个非常纯粹的技术交流的地方

具体加群方式，请参见目录菜单下的文档：《付费用户如何加群》（**购买后可见**）